

## OPTIC FLOW ESTIMATION INSIDE A BOUNDED DOMAIN

Anne Rougée\*, Bernard C. Levy\*\*, Alan S. Willsky\*\*

Laboratory for Information and Decision Systems  
M.I.T., Cambridge, MA. 02139

### Abstract

The problem of reconstructing the apparent velocity field (optic flow) in a sequence of images is formulated as a linear estimation problem. In a first stage, the case of a moving contour is considered. The optic flow along the contour is modeled as a 1-D vector Wiener process. Then, given its normal component along the contour, its tangential component is estimated by using standard 1-D smoothing techniques. In a second stage, the problem of estimating the optic flow inside a bounded domain, given an estimate on the edge and some observations inside the domain, is formulated as an estimation problem for a 2-D boundary value stochastic process. The estimator is then obtained as the solution of a system of elliptic partial differential equations with boundary conditions.

---

\* The work of this author was supported by I.N.R.I.A. (Institut National de la Recherche en Informatique et Automatique), Le Chesnay, France.

\*\* The work of these authors was supported in part by the National Science Foundation under Grant ECS-83-12921, and in part by the Army Research Office under Grant DAAG-29-84-K-0005.

## I. INTRODUCTION

The objective of this paper is to reconstruct the apparent velocity field, or optic flow, in a sequence of images. The approach that we shall follow is to construct a stochastic model for the optic flow, and then to use 1-D and 2-D estimation techniques to estimate the optic flow from available measurements. This method can be viewed as a model-based implementation of the regularization techniques which have been proposed recently [1] for ill-posed problems in computer vision.

The problem of motion estimation has been a topic of interest in image processing since the early seventies, motivated by applications such as target tracking in the military domain, or motion compensation in television images coding. Several estimation methods have already been proposed [2]-[5]. More recently, because of the appearance of new applications such as robot navigation, the objective in motion estimation has shifted from estimating the velocity field itself to acquiring some information on the observed scene.

The main feature of the motion estimation problem, by comparison with other image processing problems such as edge detection or object recognition, is that it requires explicitly the introduction of a physical model. Indeed, unlike in other applications, in motion estimation the data is not produced by a single image, but from a sequence of images. It is therefore important to use models to describe the relation existing between these successive images or, in the case of rigid motion, to describe motion parameters such as rotation and translation vectors.

The fundamental equation for motion estimation is the brightness constraint, which relates the brightness function and the velocity at any point in the image, and expresses that the brightness of a particular moving point is constant in time. It is obtained after a first order approximation of the brightness difference in time. This single scalar equation does not allow the reconstruction of both components of the optic flow, and most of the methods found in the literature for solving this equation are regularization methods, which select a particular solution by minimizing an error criterion containing a regularity constraint for the reconstructed optic flow.

One of the first and most important reconstruction methods was introduced by Horn and Schunck [2], and several extensions were developed thereafter by Cornelius and Kanade [3], Nagel [4] and others. According to this technique, the optic flow is estimated by minimizing a

criterion which includes both the average error for the brightness constraint on the whole surface of the image and a regularity constraint for the gradient of the optic flow. Using the calculus of variations, Horn and Schunck [2] obtained a solution satisfying a system of elliptic partial differential equations (more precisely, a system of coupled Poisson equations). They used the Gauss-Seidel method to solve this system iteratively, but more recently Glazer [6] and Terzopoulos [7] implemented multilevel relaxation methods, which are more efficient from a computational point of view.

Nevertheless, problems occur at discontinuities of the velocity field, due to occluding boundaries for example, which need to be treated beforehand. This has motivated researchers to develop estimation techniques where the objective is only to reconstruct the optic flow along a moving contour, instead of the whole domain, and where it is assumed that the perpendicular component of the velocity field along the contour can be computed by local methods.

Hildreth [8] has implemented a regularization method based upon the same criterion as in Horn and Schunck [2], but restricted to a contour, and where the conjugate gradient method is used for minimizing this criterion. Another method was introduced by Bouthémy [9], using a totally different point of view: in a first step, moving edges and the corresponding perpendicular velocity components are simultaneously locally detected and estimated by hypothesis testing, avoiding the use of the brightness constraint which is an approximate equation and is not valid across discontinuities. Then, in a second step the complete velocity is recovered by using a stochastic gradient algorithm along the detected contour in order to minimize the error on the perpendicular velocity component.

Although the second method is computationally superior, both of these methods are iterative. In this paper, by modeling the velocity field along the contour as a 1-D vector Wiener process, it will be shown that the reconstruction problem can be formulated as a 1-D smoothing problem, for which a number of exact, recursive solutions are available (see Ljung and Kailath [10], [11]). The main advantage of these smoothing solutions is that they are one-shot techniques, and require therefore considerably less computational effort than the two methods mentioned above.

Once the velocity field has been reconstructed along a closed contour, the next step is to estimate the field inside the bounded domain defined by this contour. In practice, this means that the detected edges have to be linked together, in order to segment the image into bounded domains which are homogeneous regions for the velocity. As was noted by Horn [5], the

image segmentation and velocity estimation problems are not independent, since in theory the image can be segmented only if the velocity field is already known. However some segmentation methods can be implemented without estimating explicitly the velocity beforehand (Bouthémy [12]).

Concerning this second problem, the approach considered here relies on the introduction of a 2-D internal stochastic model for the velocity field. The velocity field is modeled as a 2-D vector Brownian motion, and the observations are given by the constraint relating the brightness function to the velocity at any point in the image. It will be shown that this approach generalizes the one considered for the contour problem.

This paper is organized as follows. In section II we first investigate the case of a moving contour, and give a new formulation in terms of an optimal control problem or equivalently in terms of a 1-D fixed-interval smoothing problem, for which an exact solution is obtained. In section III, we consider the case of a bounded domain defined by a closed contour on which the velocity field is given (or previously estimated), and formulate this problem as a linear estimation problem for a 2-D boundary value process. This leads us to a solution satisfying to the same elliptic partial differential equations obtained by Horn and Schunck [2]. In section IV we give some experimental results, where we vary the parameters of the 1-D and 2-D stochastic models. Finally, section V contains some conclusions and some thoughts for further research.

## II. ESTIMATION OF THE VELOCITY FIELD ON A CONTOUR

In this section, we consider the case of a moving contour, whose motion has been detected beforehand from a sequence of images, and on which it is only possible to estimate locally the perpendicular component of the optic flow. We seek to reconstruct completely the optic flow along this contour, by taking in account at each point the information available over the whole contour in order to estimate the optic flow locally.

Thus let assume that we are given a contour  $C$  in an image at time  $t$ , (which means that the image brightness is constant on  $C$ ), and that this contour is parametrized as

$$C = \{ (x(s), y(s)) , s \in [0, L] \} ,$$

where  $s$  denotes the arclength, and  $x(s)$ ,  $y(s)$  are the spatial coordinates in the image plane. The apparent local velocity vector at point  $s$  on  $C$  and at time  $t$  is defined by

$$V(s) = (u, v)^T \quad (2.1)$$

with

$$u = dx/dt , \quad v = dy/dt \quad (2.2)$$

Let the image brightness at point  $(x, y)$  in the image plane at time  $t$  be denoted by  $E(x, y, t)$ . The brightness is assumed to be constant in time on the contour  $C$ , so that, taking a first order approximation of its global differential in time, we obtain the so called brightness constraint equation

$$E_x u + E_y v + E_t = 0$$

where the subscripts  $x$ ,  $y$  and  $t$  denote the partial derivatives with respect to  $x$ ,  $y$  and  $t$  respectively. This equality can be rewritten as

$$VE \cdot V = -E_t \quad (2.3)$$

where  $VE$  is the gradient of  $E(x, y, t)$  with respect to the spatial coordinates  $(x, y)$ , and where the inner product is denoted by a dot. Note that  $VE$  is

perpendicular to the contour  $C$ , since the brightness function is constant on  $C$ . It turns out from this equality that only the component of  $V$  perpendicular to the contour  $C$  can be estimated by local methods, a fact which leads us to consider global methods.

Thus, we assume that we know the perpendicular component of the velocity field on the contour. This can be expressed as

$$\mathbf{n}(s) \cdot \mathbf{V}(s) = z(s) \quad (2.4)$$

where  $\mathbf{n}(s)$  denotes the unit vector perpendicular to the contour, and where from (2.3) we note that

$$z(s) = -E_t / \|\nabla E\| \quad (2.5a)$$

and

$$\mathbf{n}(s) = (E_x, E_y)^T / \|\nabla E\| \quad (2.5b)$$

The velocity field reconstruction requires an additional constraint: we consider here a smoothness constraint, which involves the derivative of the velocity field with respect to the arclength  $s$ . This leads us to minimize the following quadratic criterion

$$J = 1/2 \int_C (a \|\mathbf{e}(s)\|^2 + \|\mathbf{dV}/\mathbf{ds}\|^2) ds \quad (2.6)$$

where

$$\mathbf{e}(s) = z(s) - \mathbf{n}(s) \cdot \mathbf{V}(s) \quad (2.7)$$

and  $a$  is a weighting factor, (see Hildreth [8]).

It turns out that this problem can be considered as an optimal tracking problem: given the state-space model

$$\begin{cases} \mathbf{dV}/\mathbf{ds} = \mathbf{U}(s) \end{cases} \quad (2.8a)$$

$$\begin{cases} V_n(s) = \mathbf{n}(s) \cdot \mathbf{V}(s) \end{cases} \quad (2.8b)$$

where  $U(s)$  is the input, we want the output signal  $V_n(s)$  to track as well as possible the desired output  $z(s)$ . The tracking error is given by

$$z(s) - V_n(s) = e(s)$$

and it is straightforward to check that it is equivalent to minimize the tracking error and to minimize the mismatch of the brightness constraint (2.3). The optimal control law for this problem minimizes the cost functional (2.6), where the term

$$\| dV/ds \|^2 = \| U(s) \|^2$$

is the control energy.

The solution of this problem is given by (Athans and Falb [13], p.793):

$$U(s) = q(s) - \Theta(s)\hat{V}(s) \quad (2.9)$$

where  $\Theta(s)$  is a real, symmetric and positive definite matrix which satisfies the Riccati matrix differential equation

$$d\Theta/ds = \Theta(s)^2 - a n(s)n(s)^T \quad (2.10)$$

with the boundary condition  $\Theta(L) = 0$ , and where the vector  $q(s)$  is the solution of the linear vector differential equation

$$dq/ds = \Theta(s)q(s) - a z(s)n(s) \quad (2.11)$$

with the boundary condition  $q(L) = 0$ . The optimal trajectory is then the solution of the linear differential equation

$$dV/ds = -\Theta(s)\hat{V}(s) + q(s) \quad (2.12a)$$

starting at the initial state

$$\hat{V}(0) = \Theta(0)^{-1}q(0) \quad (2.12b)$$

Note that this method consists of two stages: the first one is the computation of the matrix  $\Theta(s)$  and the vector  $q(s)$ , starting from  $s = L$  and going backwards, and the second is the computation of the vector  $\hat{V}(s)$ , starting from  $s = 0$  and in the forward direction.

Another point of view is to formulate the above problem in terms of a 1-D fixed-interval smoothing problem as follows. Given the state-space model

$$\begin{cases} dV/ds = U(s) \end{cases} \quad (2.13a)$$

$$\begin{cases} z(s) = n(s).V(s) + e(s) \end{cases} \quad (2.13b)$$

where  $U(s)$  and  $e(s)$  are uncorrelated white noises, with intensities  $I$  and  $1/a$  respectively, and where the observations  $z(s)$  for  $s \in [0, L]$  are given by (2.5a), we seek to compute the smoothed estimate

$$\hat{V}_S(s) = \hat{V}(s | L) = E \{ V(s) | z(\sigma), 0 \leq \sigma \leq L \} \quad (2.14)$$

According to this approach, the velocity field is modelled as a 1-D stochastic process, on which some information is given at each point of the contour by the observation equation (2.13b), which is nothing else than the brightness constraint. The estimation procedure consists in considering at each point all the available observations to compute the estimate  $\hat{V}_S(s)$ .

It was shown by Bryson and Frazier [14] that this smoothing problem is equivalent to that of minimizing the quadratic functional (2.6), and therefore a solution of this problem is given by (2.9)-(2.12). However, several solutions of the fixed-interval smoothing problem (such as the Mayne-Fraser two-filter formula, the innovations method...) have been proposed over the years (see Mayne [15], Fraser [16], Ljung and Kailath [10], [11]), and any of these methods can be used for solving the above problem.

In the smoothing context, the solution (2.9)-(2.12) can be identified as the Rauch-Tung-Striebel [17] implementation of the smoother, where if  $\hat{V}_b(s)$  and  $P_b(s)$  denote respectively the backwards filtered estimate and filtering error variance of  $V(s)$ , we can identify

$$\theta(s) = P_b^{-1}(s) \quad , \quad q(s) = P_b^{-1}(s) \hat{V}_b(s) \quad (2.15a)$$

and

$$\hat{V}(s) = \hat{V}_S(s) \quad (2.15b)$$

The variance of the smoothed estimate is given by



$$P_S^{-1}(s) = P_f^{-1}(s) + P_b^{-1}(s) \quad (2.16)$$

where  $P_f(s)$  is the forwards filtering error variance. The inverses of the forwards and backwards filtering error variances satisfy Riccati equations of the form (2.10), with initial conditions at  $s = 0$  and  $s = L$  respectively.

### III. ESTIMATION OF THE VELOCITY FIELD IN A BOUNDED DOMAIN

We now investigate the problem of reconstructing the velocity field in a bounded domain given its value on the edge. The formulation adopted here is in terms of a linear estimation problem for a 2-D boundary value stochastic process, following the approach of Adams, Willsky and Levy [18], which relies on the theory of complementary models.

Thus, we assume we are given a bounded completely connected region  $D$  of the image plane at time  $t$ , with a smooth boundary  $C$ , on which the optic flow is known, or has been estimated beforehand by using the method of section II. In fact, such domains are not usually found in real images, but a preliminary edge detection scheme may provide a segmentation of the image into regions which may reasonably be assumed to be homogeneous with regard to the velocity.

The velocity field  $V$  at time  $t$  is defined as in (2.1)-(2.2), and the velocity field "gradient" is the vector containing all the spatial partial derivatives of its components, which is defined by

$$LV = (u_x, u_y, v_x, v_y)^T = \begin{bmatrix} \nabla u \\ \nabla v \end{bmatrix} \quad (3.1)$$

The differential operator  $L$  can be formally expressed as

$$L = (I \otimes \nabla) \quad (3.2)$$

where  $\otimes$  denotes the Kronecker product, which indicates that the gradient operates on each component of  $V$ , but for simplicity it will also be denoted by  $\nabla$  in the following.

The function  $E(x,y,t)$  still denotes the image brightness at time  $t$ . The brightness at a particular point  $(x,y) \in D$  is assumed to be constant in time so that the brightness equation (2.3) still holds.

The process  $V$  to be estimated is then formally defined by the state-space model

$$\begin{cases} \nabla V = VB \end{cases} \quad (3.3a)$$

$$\begin{cases} z(x,y) = H(x,y)V(x,y) + e(x,y) \end{cases} \quad (3.3b)$$

where  $e(x,y)$  is a white noise with intensity  $1/a$ ,  $B(x,y)$  is a Brownian motion with two parameters, so that  $VB$  is a vector white noise process with intensity  $I$ , uncorrelated with  $e$ . The matrix  $H$  is given by

$$H(x,y) = (E_x, E_y) \quad (3.4)$$

and the set of observations is given by

$$z(x,y) = -E_t, \quad (x,y) \in D \quad (3.5)$$

The boundary condition is expressed as

$$V(x,y) = V_C(x,y), \quad (x,y) \in C \quad (3.6)$$

where  $V_C$  is gaussian with mean  $\hat{V}_S$  and covariance  $P_S$ , and where  $\hat{V}_S$  and  $P_S$  denote respectively the smoothed estimate and error variance obtained by estimating the velocity field  $V$  on the contour  $C$ . This boundary condition is not particularly constraining, since the absence of information on some part of the contour may be modeled by selecting an infinite variance at points where the velocity is unknown. This observation is quite useful in practice, since the domain that we consider is usually obtained by linking together several unconnected space-time edges which have been identified by using an edge detection procedure. In this case no information is available on the linking segments, and on these segments we can therefore take  $P_S = \infty$ .

It is worth noting that the above model is a generalization of the one introduced for the contour problem. More precisely, (3.3a) is a generalization of (2.13a) since we have the following relation on the contour  $C$

$$\begin{aligned} dV/ds &= (u_x dx/ds + u_y dy/ds, v_x dx/ds + v_y dy/ds)^T \\ &= (t(s) \cdot \nabla u, t(s) \cdot \nabla v)^T \\ &= (I \otimes t(s)^T) \nabla V \end{aligned} \quad (3.7)$$

where  $t(s)$  denotes the unit vector tangent to the contour  $C$  i.e.

$$t(s) = (dx/ds, dy/ds)^T \quad (3.8)$$

Similarly, the 1-D white noise process  $U(s)$  appearing on the right hand side of (2.13a) can be expressed as

$$U(s) = dB/ds = (I \otimes t(s)^T) VB \quad (3.9)$$

where we have used here the fact that the restriction of the 2-D Brownian process  $B(x,y)$  to the boundary  $C$  gives rise to a 1-D Brownian process  $B(s) = B(x(s),y(s))$ .

Thus, the models (2.13) and (3.3) of the vector field  $V$  on the contour  $C$  and inside the domain  $D$  are perfectly consistent, and correspond to assuming that  $V(x,y)$  is a 2-D Brownian process.

To estimate  $V(x,y)$  we will now follow the procedure introduced by Adams and al. [18]. From the classical Green's identity

$$\int_D \nabla u \cdot \mathbf{W} \, dx \, dy = \int_D u (-\nabla \cdot \mathbf{W}) \, dx \, dy + \int_C u (\mathbf{n}(s) \cdot \mathbf{W}) \, ds \quad (3.10)$$

for any scalar process  $u$  and vector process  $\mathbf{W}$ , where  $\nabla$  denotes the divergence operator, and  $\mathbf{n}(s)$  the unit vector perpendicular to the contour  $C$ , we can derive the following Green's identity satisfied by the  $L^*$  operator defining our system

$$\int_D LV \cdot \mathbf{W} \, dx \, dy = \int_D V \cdot L^* \mathbf{W} \, dx \, dy + \int_C V \cdot N \mathbf{W} \, ds \quad (3.11)$$

where the formal adjoint of the differential operator  $L$ , denoted by  $L^*$ , and the operator  $N$  are given by

$$L^* = -(I \otimes \nabla) \quad (3.12)$$

$$N = (I \otimes \mathbf{n}(s)) \quad (3.13)$$

Using the theory of complementary models, the estimator is then found as the solution of the following linear system (Adams et al. [18])

$$LV = \mathbf{W} \quad (3.14a)$$

$$a H^T H V + L^* \mathbf{W} = a H^T z \quad (3.14b)$$

where  $\mathbf{W}$  is the estimate of the complementary process of  $\mathbf{V}$ . Eliminating  $\mathbf{W}$  in (3.14), we obtain

$$\mathbf{a} \mathbf{H}^T \mathbf{H} \mathbf{V} + \mathbf{L}^T \mathbf{L} \mathbf{V} = \mathbf{a} \mathbf{H}^T \mathbf{z} .$$

This equation may be expressed more explicitly by noting that

$$\begin{aligned} \mathbf{L}^T \mathbf{L} &= -(\mathbf{I} \otimes \nabla \cdot)(\mathbf{I} \otimes \nabla) \\ &= -(\mathbf{I} \otimes \Delta) \end{aligned} \quad (3.15)$$

where  $\Delta$  denotes the Laplacian operator, and hence we obtain

$$-(\mathbf{I} \otimes \Delta) \mathbf{V} = \mathbf{a} \mathbf{H}^T (\mathbf{z} - \mathbf{H} \mathbf{V}) . \quad (3.16)$$

From (3.4)-(3.5) this system is equivalent to the following system of partial differential equations

$$\Delta u = \mathbf{a} E_x (E_t + E_x u + E_y v) \quad (3.17a)$$

$$\Delta v = \mathbf{a} E_y (E_t + E_x u + E_y v) . \quad (3.17b)$$

These equations are coupled Poisson equations. They are elliptic, i.e. non causal, and can be solved iteratively by using a discrete approximation of the Laplacian.

Using the formula found in Adams et al. [18], the boundary condition is given by

$$0 = \mathbf{P}_S^{-1} (\mathbf{V} - \hat{\mathbf{V}}_S) + \mathbf{N} \mathbf{W} \quad \text{on } C . \quad (3.18)$$

According to (3.13) and (3.14a), this can be rewritten as

$$\mathbf{V} = \hat{\mathbf{V}}_S - \mathbf{P}_S (\mathbf{I} \otimes \mathbf{n}^T) \nabla \mathbf{V} \quad \text{on } C , \quad (3.19)$$

or, in terms of the components of  $\mathbf{V}$ , as

$$\begin{bmatrix} \partial u \\ \partial v \end{bmatrix} + \mathbf{P}_S \begin{bmatrix} \partial u / \partial n \\ \partial v / \partial n \end{bmatrix} = \hat{\mathbf{V}}_S \quad \text{on } C , \quad (3.20)$$

where  $\partial u / \partial n$  and  $\partial v / \partial n$  denote the normal derivatives of  $u$  and  $v$ , respectively.

Two particular cases are of interest:

a)  $P_S = 0$ . In this case, the velocity is known exactly on the boundary. Then the boundary condition is simply a Dirichlet type condition:

$$V = \hat{V}_S \text{ on } C . \quad (3.21)$$

b)  $P_S = \infty$ , i.e. no information is given on the boundary. Then the boundary condition is a Neumann type condition:

$$\partial V / \partial n = 0 \text{ on } C . \quad (3.22)$$

In the general case (3.20), the variance matrix  $P_S$  represents the variation of  $V$  on  $C$  which is allowed around the initial estimate  $\hat{V}_S$ .

Note that the equations (3.17) are the same as those obtained by Horn and Schunck [2] by using the calculus of variations for minimizing the quadratic criterion

$$J = 1/2 \int_D ( a \| z - HV \|^2 + \| \nabla V \|^2 ) dx dy . \quad (3.23)$$

This is not really surprising since we know that the smoothing problem (3.3)-(3.4) is equivalent to that of minimizing the quadratic criterion (3.23).

In Horn and Schunck [2], the Gauss-Seidel method was used to solve the system of equations (3.17). However, one can expect that faster methods such as overrelaxation or multigrid methods will provide better results. Although the use of such heavy PDE solvers may appear inappropriate in image processing, where fast solutions are usually desired, it is useful to keep in mind the fact that the solutions that we seek need not be particularly accurate. The number of relevant digits required is much smaller than in typical PDE applications, and if we use very efficient overrelaxation or multigrid PDE solvers, only few iterations will be needed to obtain a good estimate of the optic flow. In addition, both overrelaxation and multigrid methods may be implemented in parallel on special purpose computers, so that speed is unlikely to limit the applicability of the optic flow estimation procedure described above.

## IV. EXPERIMENTAL RESULTS

### IV.1. Contour case

The results presented below have been obtained from synthetic data. In the two examples that we consider in Figs. 1-4, the contour is polygonal and is generated by linking a finite number of points and discretizing the  $x$  and  $y$  variables along each edge accordingly. The contour is displaced by applying a rigid body motion in the image plane, and the perpendicular component of the velocity along the contour is computed from the motion parameters. Figs. 1 and 2 correspond to the case of a translation and Figs. 3 and 4 to a rotation. In these figures, all points are represented.

The algorithm described in (2.10)-(2.12) is used for computing the estimate on these contours. To select a value for the parameter  $a$  appearing in the algorithm, we note from (2.13b) that  $a$  is the inverse of the measurement noise intensity. In the case considered here, since no noise has been added to the data, the measurement noise consists only of the higher order terms in the first-order difference approximation of the brightness constraint. Thus, the measurement noise should not be taken too large and, as shown in Figs. 3 and 4, experiments indicate that the best results are obtained when  $a$  is selected between 0.5 and 1.

Our results have several interesting features. First, in the translation case, since the optic flow is constant along the contour, the estimate is equal to the true value for any value of  $a$ , as indicated in Figs. 1 and 2. The results obtained in the rotation case are quite satisfying, except for small values of  $s$  and when only few points are used to discretize the contour, as in Fig. 4d. This artifact is a consequence of the fact that the smoothing procedure that we have used does not take into account the observation that the contour is closed. Thus, in the vicinity of  $s = 0$  the information provided by the measurements around  $s = L$  is lost, even though the corresponding points along the contour are very close to each other. One way to overcome this difficulty would be to replace the initial condition for the state-space model (2.13) by a periodic boundary condition of the type

$$V(0) = V(L) \quad (4.1)$$

In this case the smoothing problem is slightly more difficult, since  $V(s)$  is now a two-point boundary value stochastic process, but a recursive solution of this problem in terms of a two-filter formula was proposed by Adams et al. [18]. This solution is currently being implemented.

## IV.2 Domain case

We present here some results where the domain that we consider is either a rectangle (Figs. 5-6) or a convex polygon (Figs. 7-8). In both cases two images are generated, which correspond to two consecutive positions of the rigid object moving in the image plane. The motion is either a translation in Figs. 5-5' and 7-7', or a rotation in Figs. 6-6' and 8-8'. The intensity function inside the domain is assumed to be either sinusoidal in  $x$  and  $y$  (one period, Figs. 5-8), or linear in  $x$  and  $y$  (Figs. 5'-8'). The noise is added to the intensity gradient and its time derivative which are computed numerically from the two images, and the velocity is estimated on the boundary as previously described.

To solve the Poisson equations (3.17), a local relaxation procedure developed by Kuo, Levy and Musicus [19] was implemented. The main feature of this method is that it allows the optimal relaxation parameter to be space dependent, and it is therefore very convenient for space-variant PDEs such as (3.17). To implement this method, the domain and its boundary were discretized on a uniform grid with a grid spacing equal to  $h$ . For clarity in the rectangle and polygon examples of Figs. 5-6 and Figs. 7-8, only one point in 10 (respectively one point in 5) have been represented in each direction. In both cases the exact contour of the domain is also depicted. Then, the local relaxation procedure, which is based on a red/black or checkerboard partition of the domain, can be expressed as follows.

For each iteration  $n$ :

- Red points ( $i+j$  is even):

$$u_{i,j}^{(n+1)} = (1 - w_{i,j}) u_{i,j}^{(n)} \quad (4.2)$$

$$+ w_{i,j} d_{i,j}^{-1} (u_{i-1,j}^{(n)} + u_{i+1,j}^{(n)} + u_{i,j-1}^{(n)} + u_{i,j+1}^{(n)}) \\ - a^2 h E_x (E_y v_{i,j}^{(n)} + E_t)$$

$$v_{i,j}^{(n+1)} = (1 - w'_{i,j}) v_{i,j}^{(n)} \quad (4.3)$$

$$+ w'_{i,j} d'_{i,j}^{-1} (v_{i-1,j}^{(n)} + v_{i+1,j}^{(n)} + v_{i,j-1}^{(n)} + v_{i,j+1}^{(n)}) \\ - a^2 h E_y (E_x u_{i,j}^{(n+1)} + E_t)$$



- Black points ( $i+j$  is odd):

$$u_{i,j}^{(n+1)} = (1 - w_{i,j}) u_{i,j}^{(n)} \quad (4.4)$$

$$+ w_{i,j} d_{i,j}^{-1} (u_{i-1,j}^{(n+1)} + u_{i+1,j}^{(n+1)} + u_{i,j-1}^{(n+1)} + u_{i,j+1}^{(n+1)}) \\ - a^2 h E_x (E_y v_{i,j}^{(n+1)} + E_t)$$

$$v_{i,j}^{(n+1)} = (1 - w'_{i,j}) v_{i,j}^{(n)} \quad (4.5)$$

$$+ w'_{i,j} d'_{i,j}^{-1} (v_{i-1,j}^{(n+1)} + v_{i+1,j}^{(n+1)} + v_{i,j-1}^{(n+1)} + v_{i,j+1}^{(n+1)}) \\ - a^2 h E_y (E_x u_{i,j}^{(n+1)} + E_t)$$

In the above equations the coefficients  $d_{i,j}$  and  $d'_{i,j}$  are given by

$$d_{i,j} = 4 + a h^2 E_x^2 \quad (4.6)$$

$$d'_{i,j} = 4 + a h^2 E_y^2 \quad (4.7)$$

and the local relaxation  $w_{i,j}$  and  $w'_{i,j}$  are given by

$$w_{i,j} = 2(1 + \sqrt{1 - r_{i,j}^2})^{-1} \quad (4.8)$$

$$w'_{i,j} = 2(1 + \sqrt{1 - r'_{i,j}^2})^{-1} \quad (4.9)$$

where in the rectangle case, corresponding to  $N=K \times L$  points, the optimal choice for the parameters  $r_{i,j}$  and  $r'_{i,j}$  is the following

$$r_{i,j} = 2 d_{i,j}^{-1} (\cos(\pi/(K+1)) + \cos(\pi/(L+1))) \quad (4.10)$$

$$r'_{i,j} = 2 d'_{i,j}^{-1} (\cos(\pi/(K+1)) + \cos(\pi/(L+1))) \quad (4.11)$$

and in the polygon case we have chosen

$$r_{i,j} = 4 d_{i,j}^{-1} \cos(\pi/\partial) \quad (4.12)$$

$$r'_{i,j} = 4 d'_{i,j}^{-1} \cos(\pi/\partial) \quad (4.13)$$

where  $\delta$  is the diameter of the domain considered, in number of points.

The boundary condition has been taken as a Dirichlet type condition. The theoretical study made in Kuo et al. [19] shows that this iterative method converges for a number of iterations proportional to the square root of the number of points  $N$  in the domain, instead of  $N$  for the Gauss-Seidel method.

The results obtained are very satisfactory. In the rectangle case (Figs. 5-6), where we have used the optimal relaxation parameters, and where the boundary value has been set equal to the true value, a good estimate is obtained after only 25 iterations for almost 4000 points. In the second example (Figs. 7-8), the boundary value has been taken equal to the estimate obtained by using the contour case method, and we obtained a good estimate after only 15 or 20 iterations for 1121 points.

Finally we present now the relative errors corresponding to the results shown in Figs. 5-8, and for comparison those obtained with the same number of iterations but with a weighting factor  $\alpha$  equal to zero.

	Rectangle				Polygone			
	Translation		Rotation		Translation		Rotation	
a -	0	0.008	0	0.003	0	0.0015	0	0.0007
relative error	17.8 %	9 %	12 %	10 %	8.9 %	6.9 %	8.16 %	7.96 %

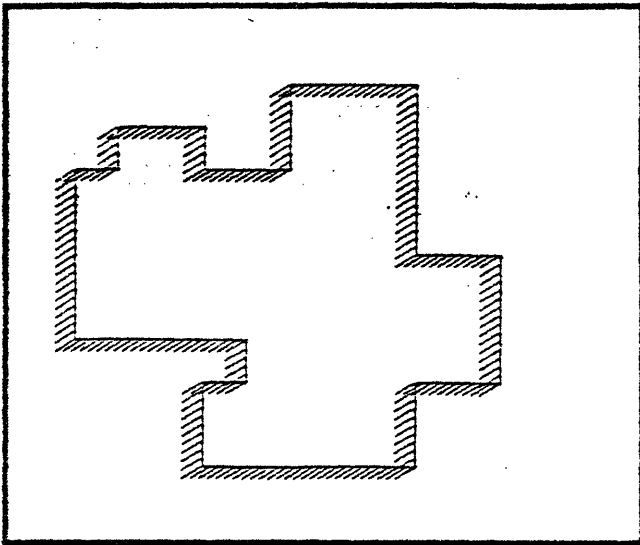


Figure 1.a: True velocity

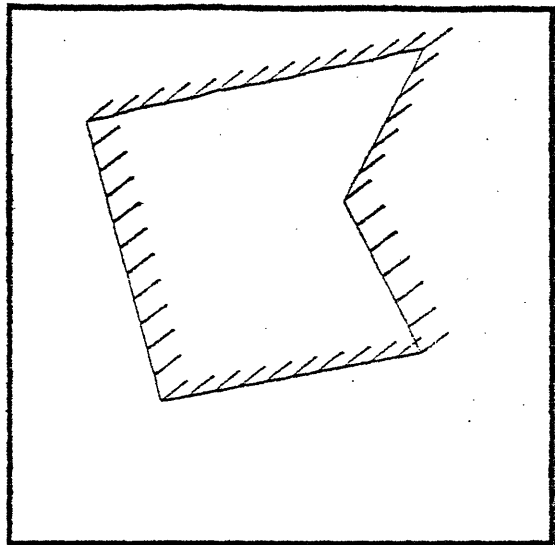


Figure 2.a: True velocity

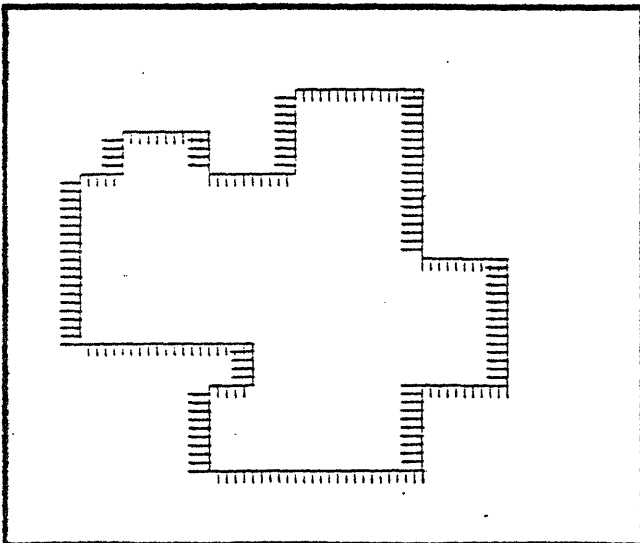


Figure 1.b: Perpendicular component

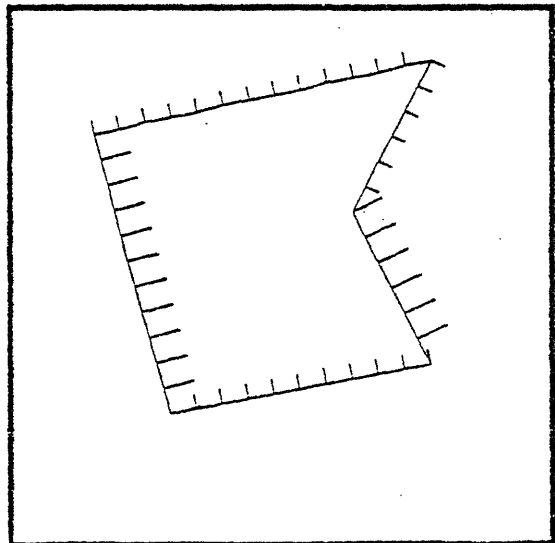


Figure 2.b: Perpendicular component

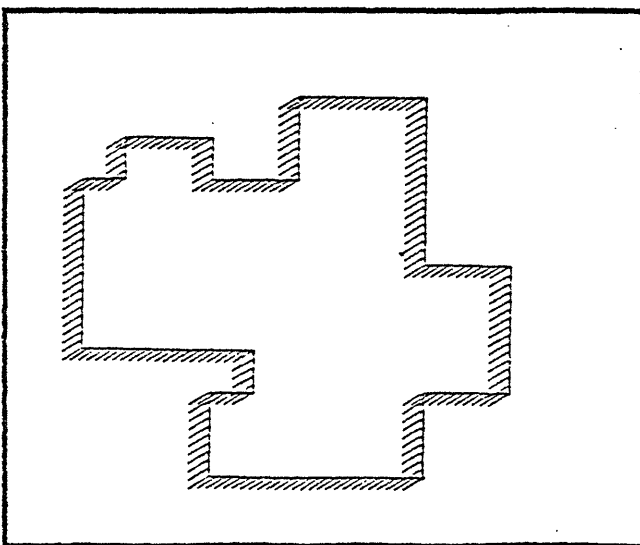


Figure 1.c: Estimated velocity

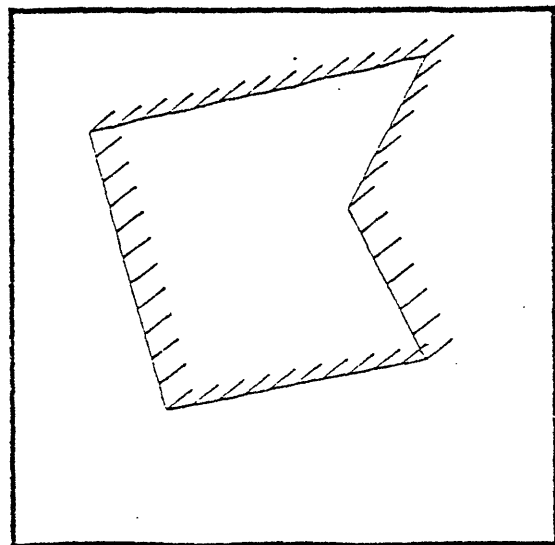


Figure 2.c: Estimated velocity

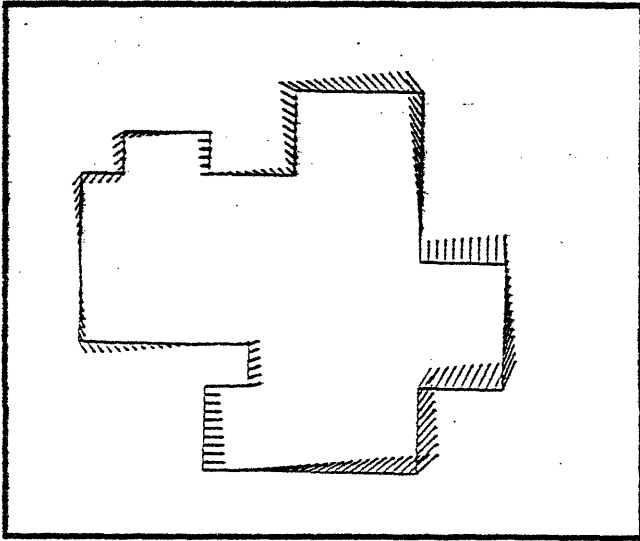


Figure 3.a: True velocity

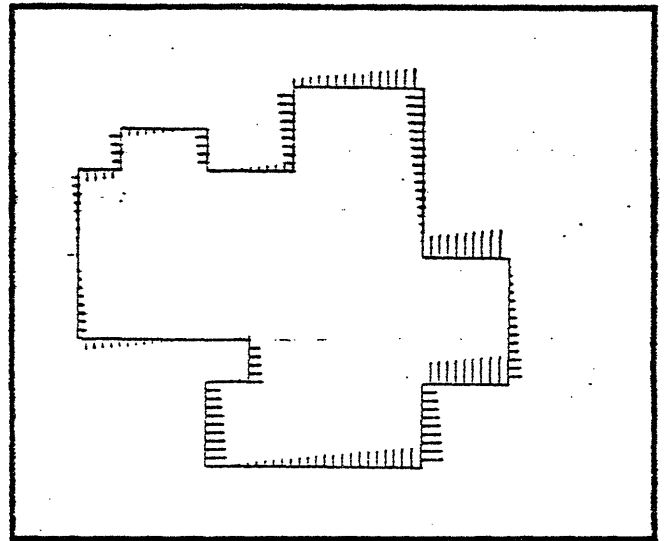


Figure 3.b: Perpendicular component

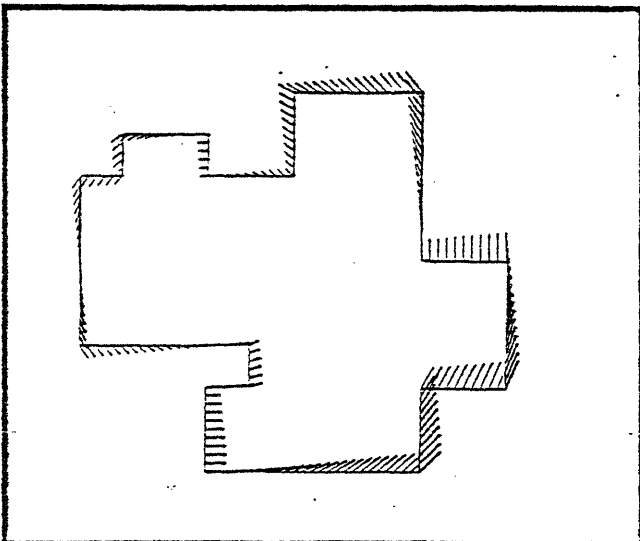


Figure 3.c: Estimated velocity,  $a=0.95$

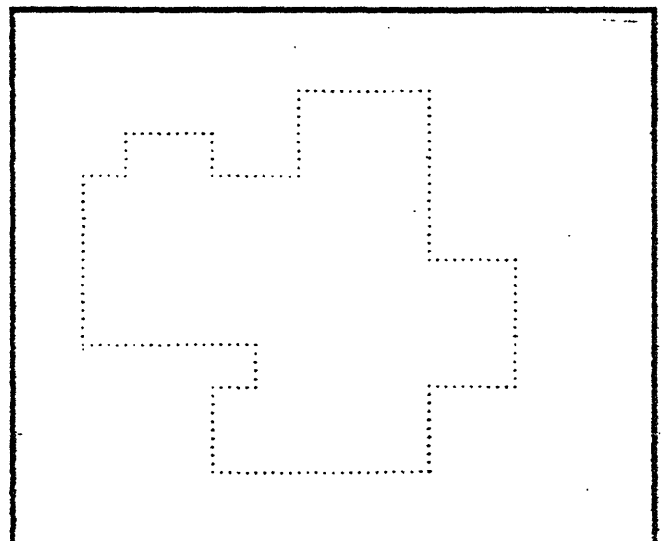


Figure 3.d: Estimation error,  $a=0.95$

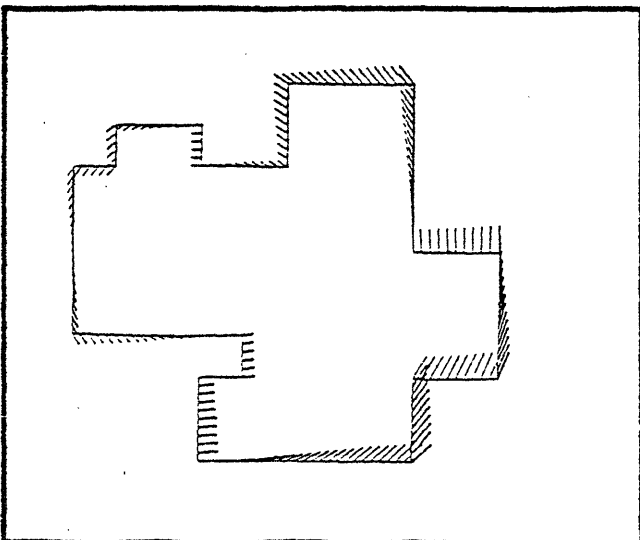


Figure 3.e: Estimated velocity,  $a=0.1$

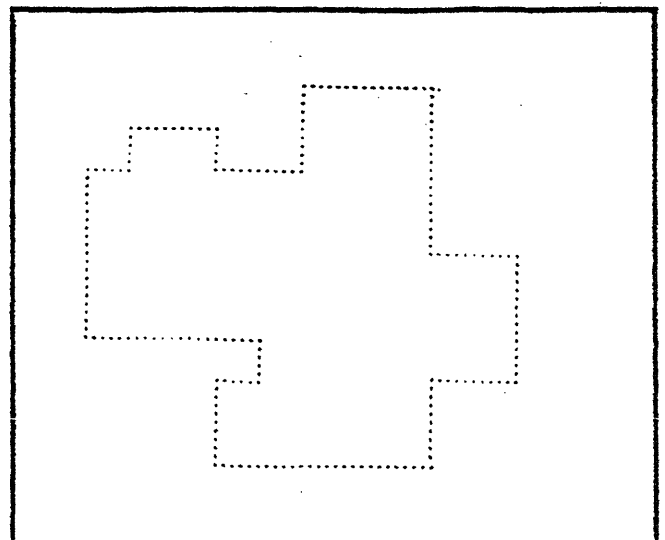


Figure 3.f: Estimation error,  $a=0.1$

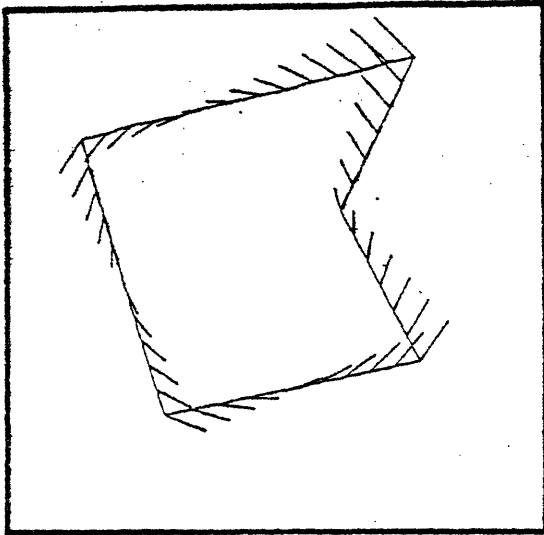


Figure 4.a: True velocity

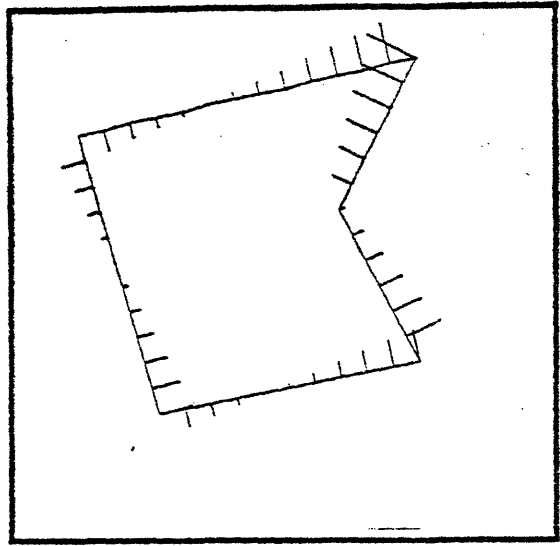


Figure 4.b: Perpendicular component

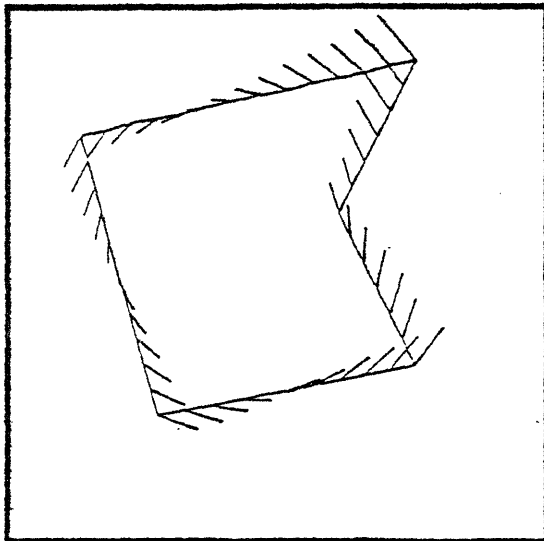


Figure 4.c: Estimated velocity,  $a=0.8$

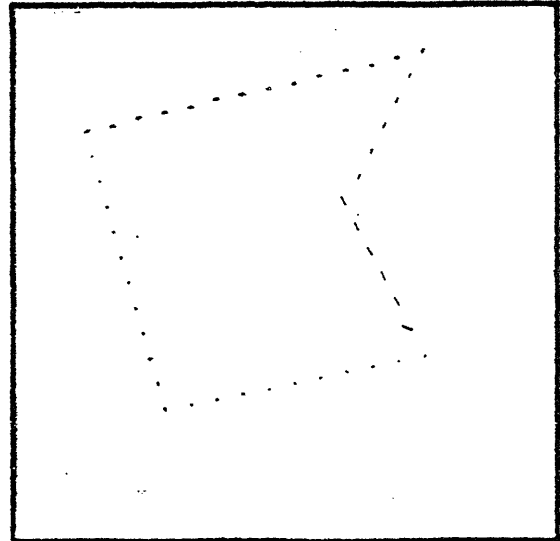


Figure 4.d: Estimation error,  $a=0.8$

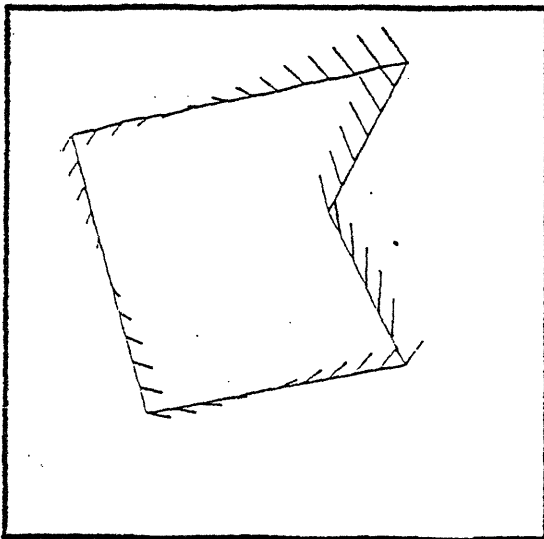


Figure 4.e: Estimated velocity,  $a=0.1$

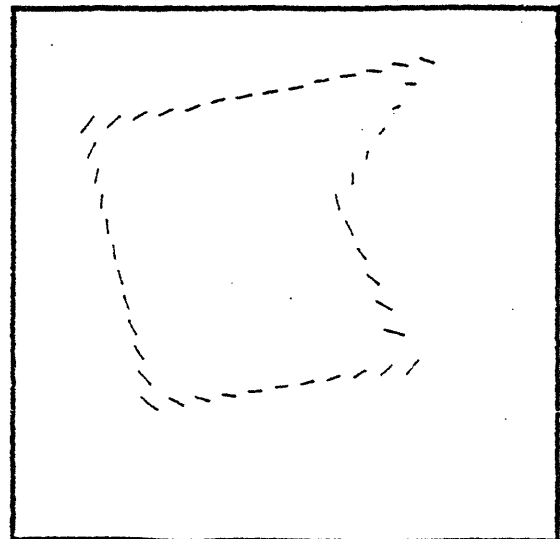


Figure 4.f: Estimation error,  $a=0.1$

## Translation

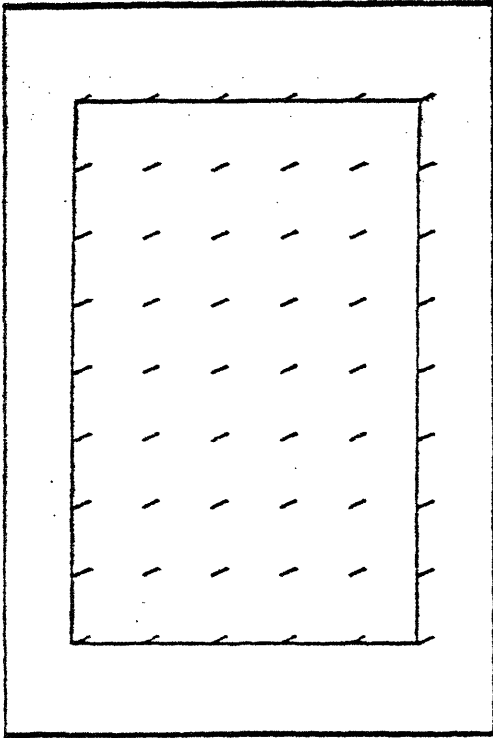


Fig. 5.a: True velocity

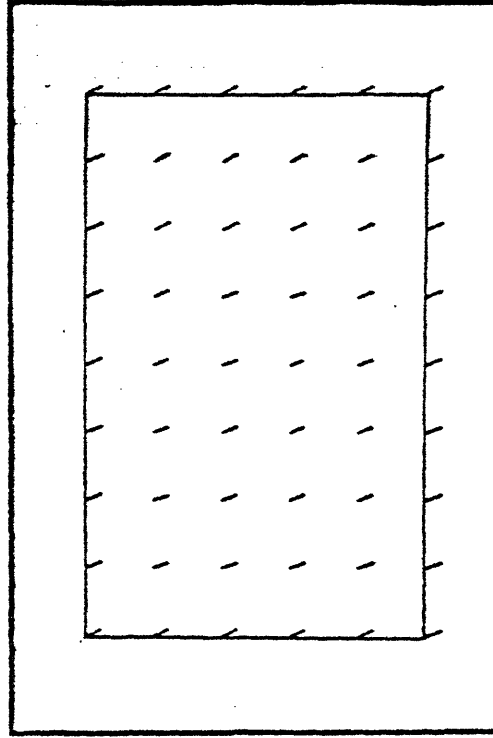
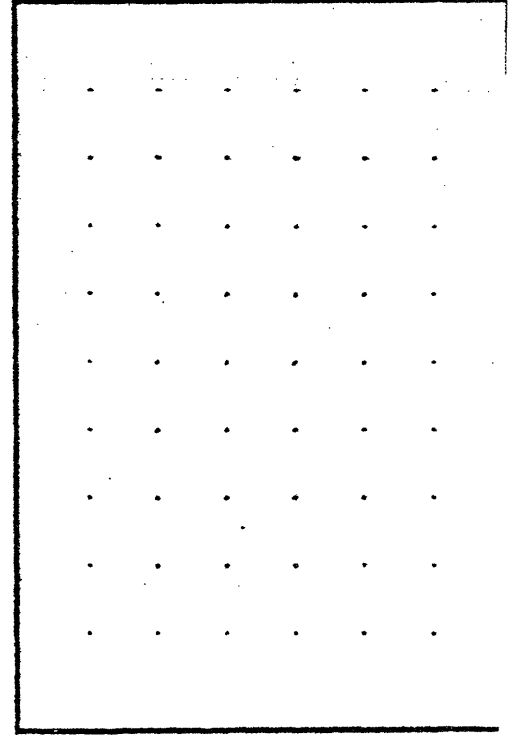
Fig. 5.b: Estimated velocity  
25 iterations,  $\alpha = 0.008$ 

Fig. 5.c: Estimation error

## Rotation

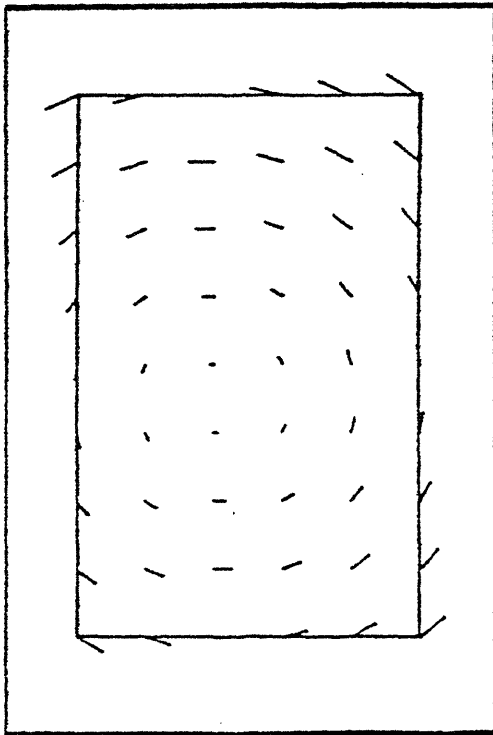


Fig. 6.a: True velocity

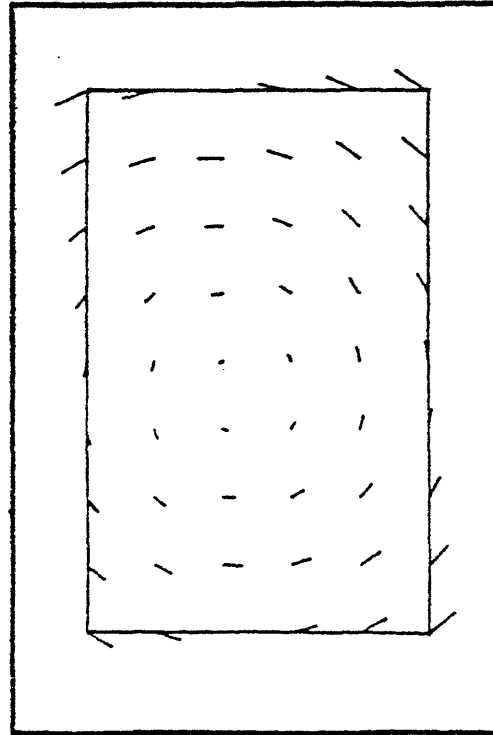
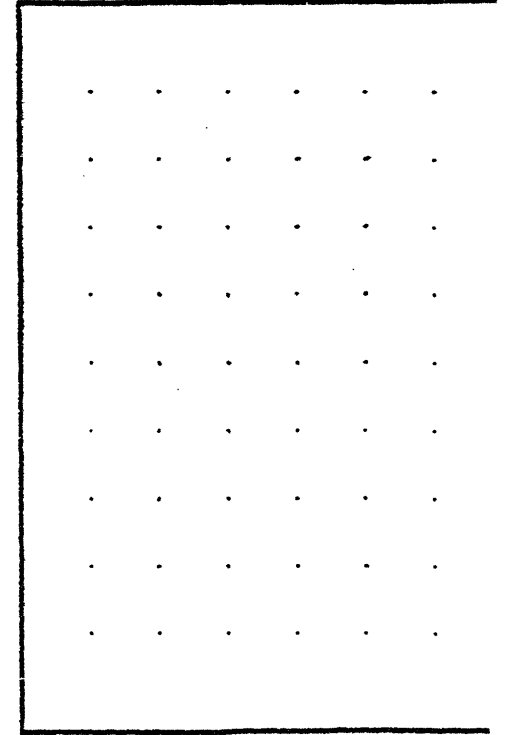
Fig. 6.b: Estimated velocity  
25 iterations,  $\alpha = 0.003$ 

Fig. 6.c: Estimation error

## Translation

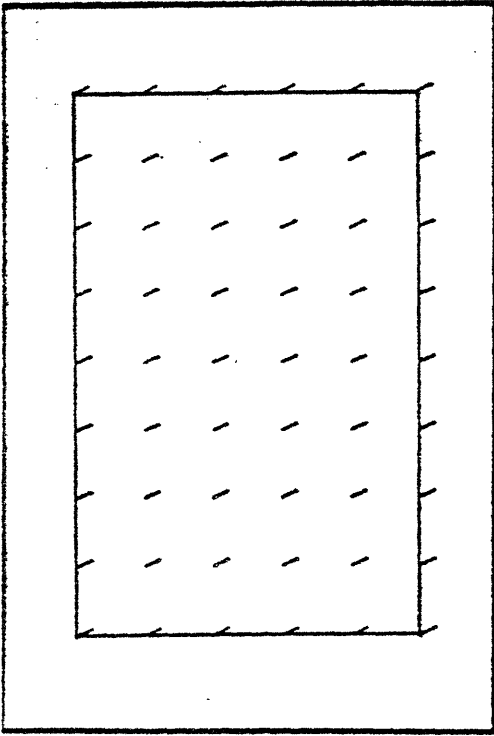


Fig. 5.a: True velocity

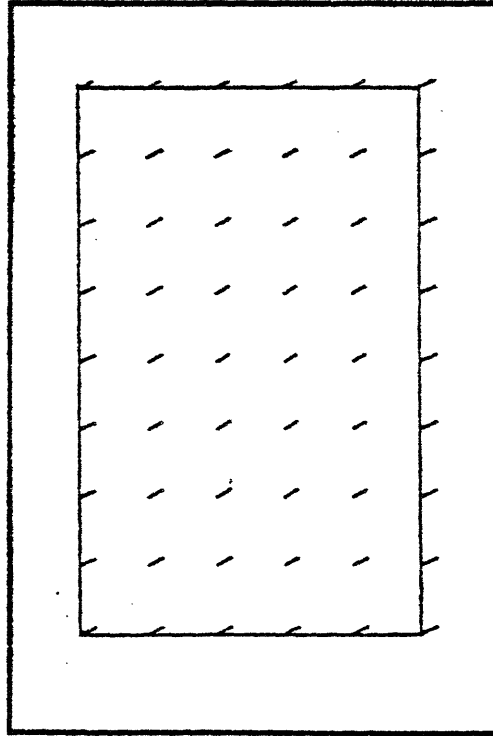
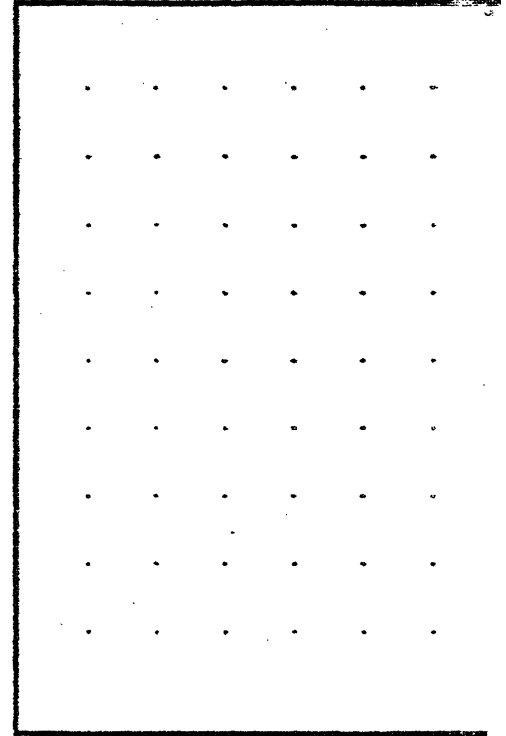
Fig. 5.b: Estimated velocity  
25 iterations,  $\alpha = 0.025$ 

Fig. 5.c: Estimation error

## Rotation

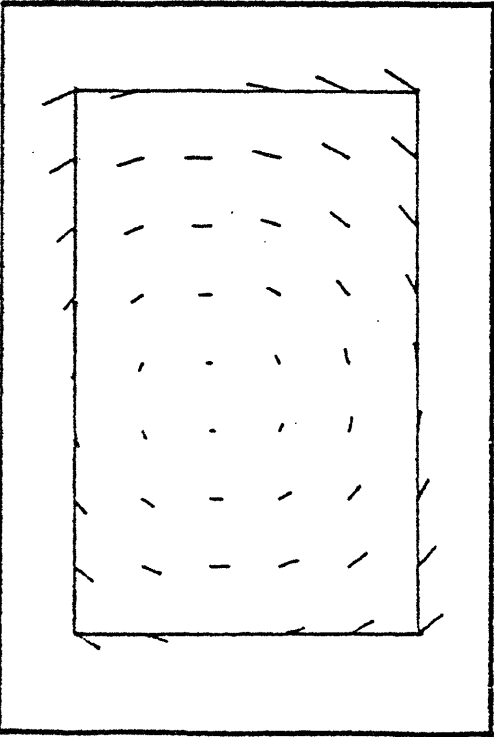


Fig. 6.a: True velocity

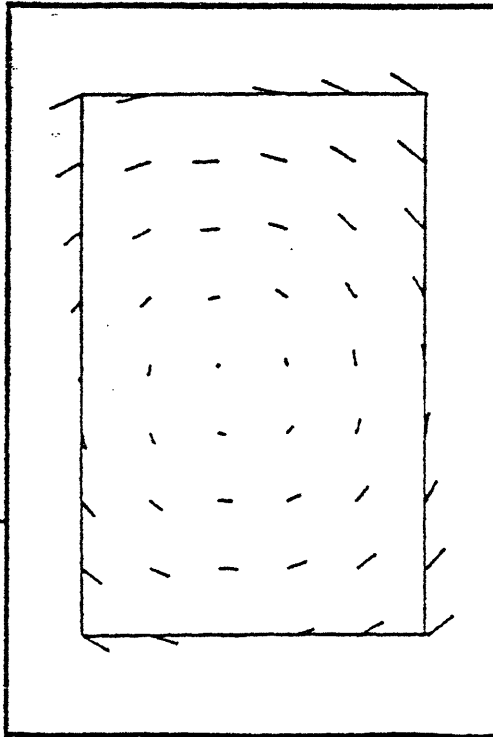
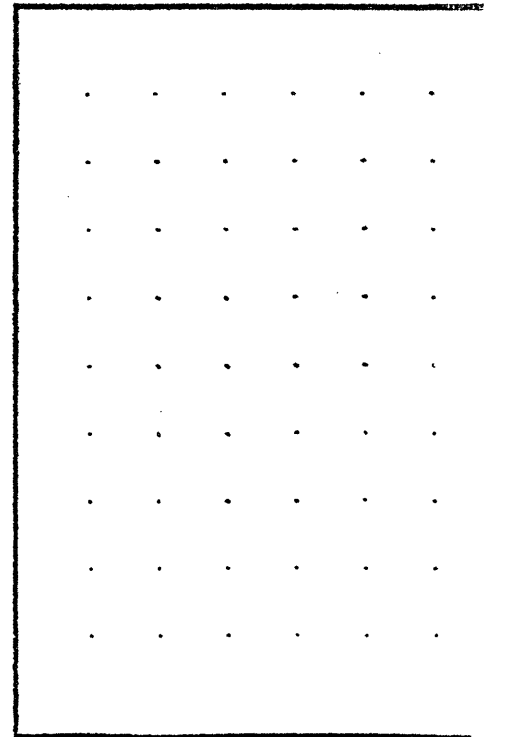
Fig. 6.b: Estimated velocity  
25 iterations,  $\alpha = 0.001$ 

Fig. 6.c: Estimation error

## Translation

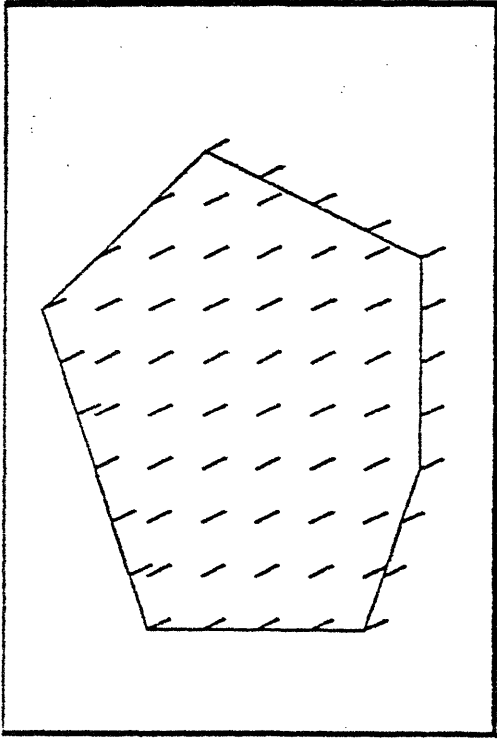


Fig. 7.a: True velocity

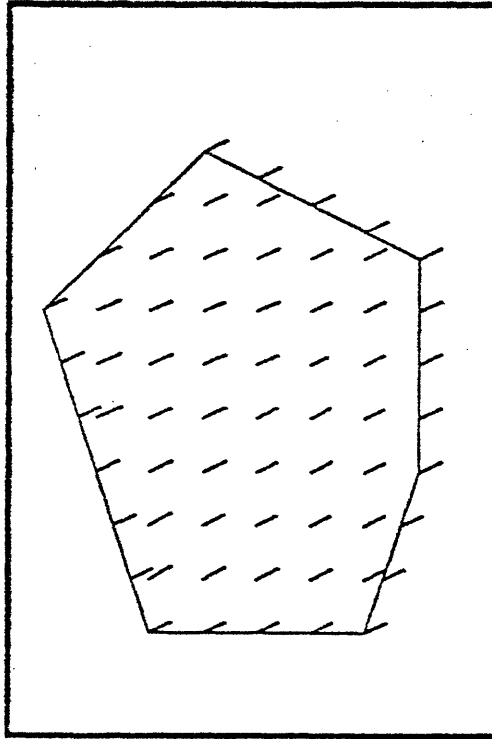
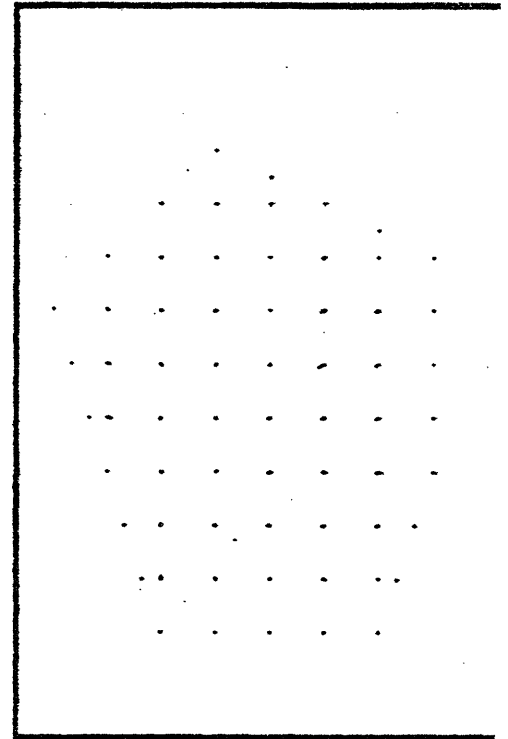
Fig. 7.b: Estimated velocity  
15 iterations,  $\alpha = 0.0015$ 

Fig. 7.c: Estimation error

## Rotation

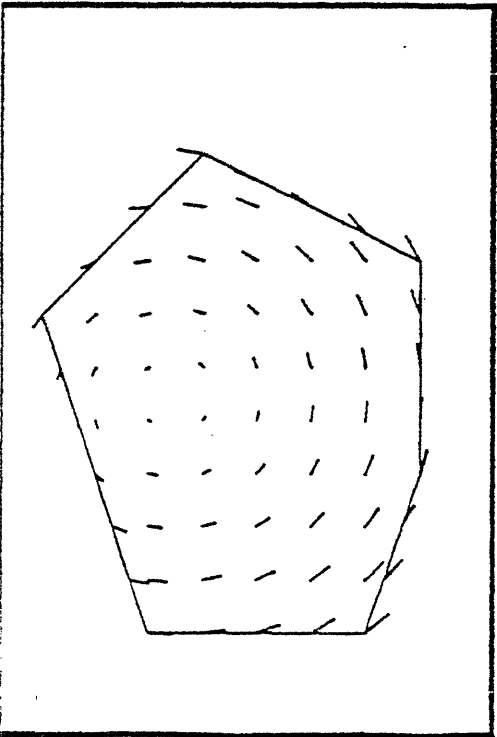


Fig. 8.a: True velocity

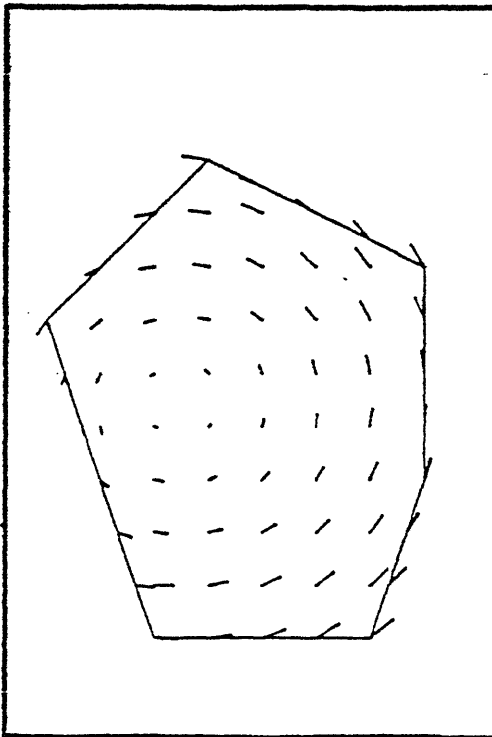
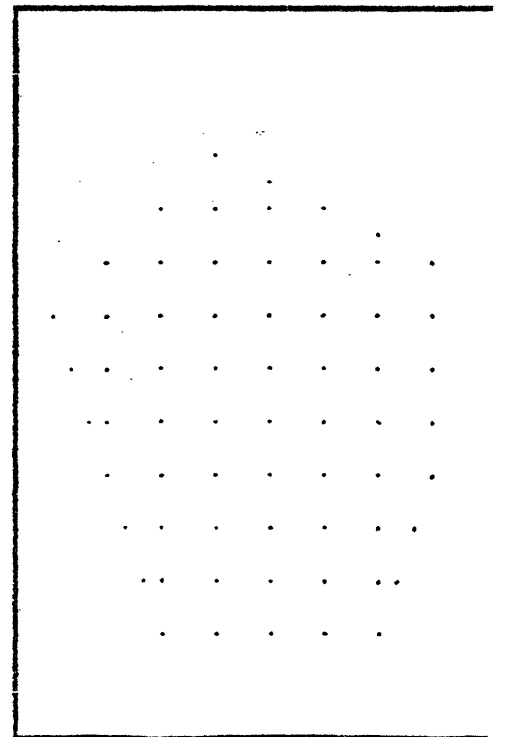
Fig. 8.b: Estimated velocity  
20 iterations,  $\alpha = 0.007$ 

Fig. 8.c: Estimation error



## Translation

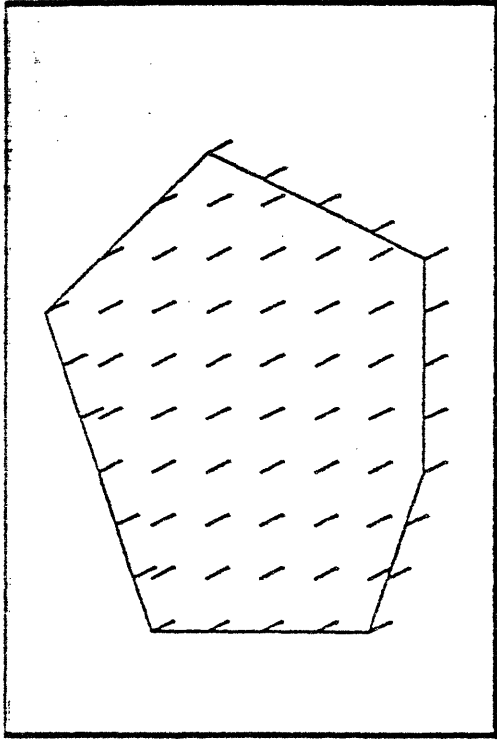


Fig. 7'a: True velocity

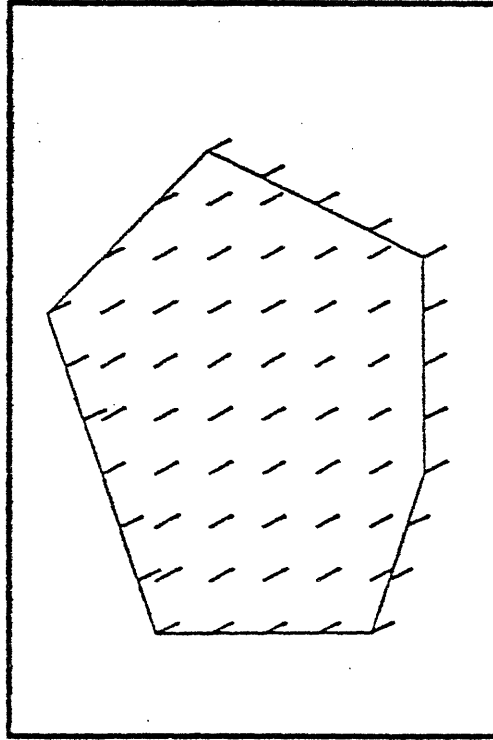
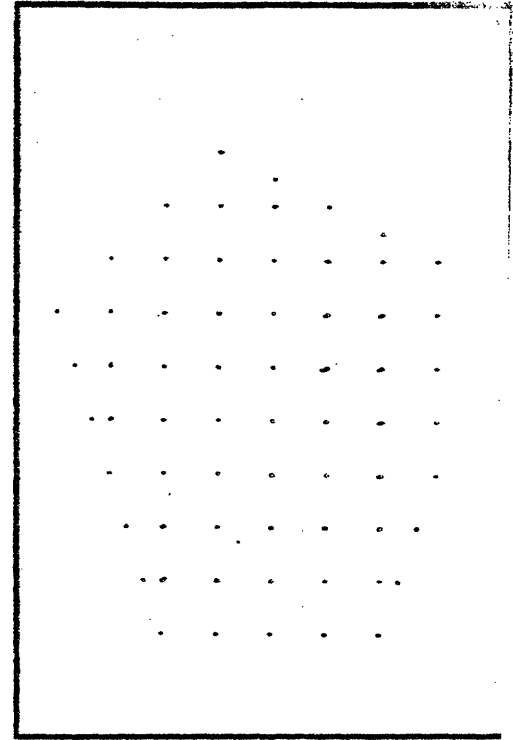
Fig. 7'b: Estimated velocity  
15 iterations,  $\alpha = 0.004$ 

Fig. 7'c: Estimation error

## Rotation

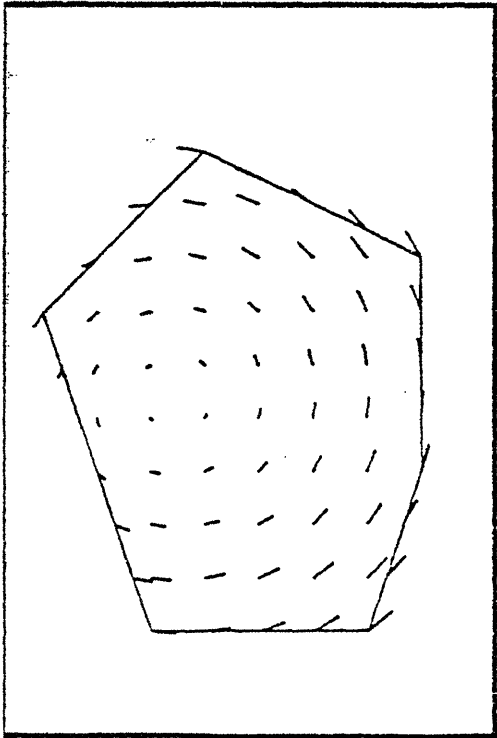


Fig. 8'a: True velocity

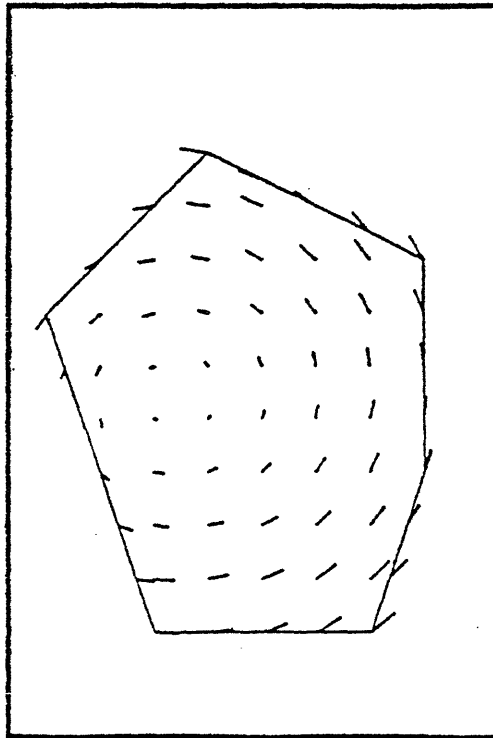
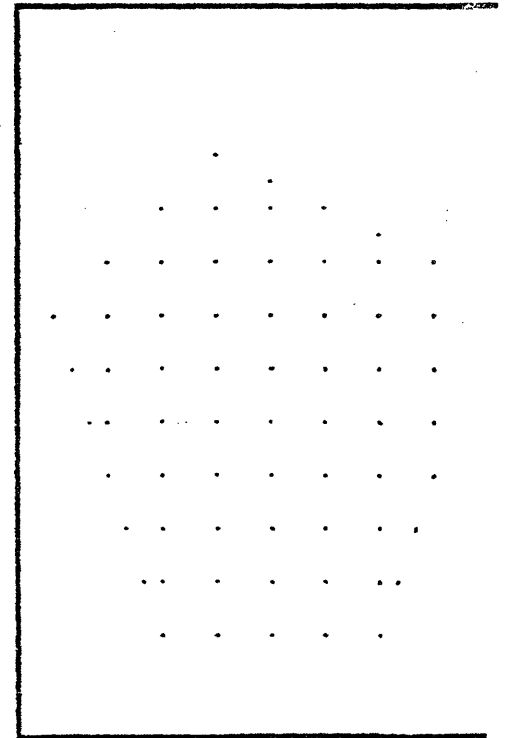
Fig. 8'b: Estimated velocity  
20 iterations,  $\alpha = 0.001$ 

Fig. 8'c: Estimation error

## V. CONCLUSION

We have derived first a method for estimating the optic flow along a contour which relies on classical 1-D smoothing techniques. This method solves exactly the estimation problem formulated in (2.8), and experimental results confirm that by using this approach a highly accurate estimate of the optic flow can be obtained very quickly. In order to compute this estimate the contour is scanned only twice, once forwards and once backwards, whereas in the iterative conjugate gradient or adaptive stochastic gradient techniques of Hildreth [8] and Bouthémy [9], the contour needs to be scanned a number of times. In addition, we expect that even better results can be obtained when, following Adams et al. [18], a periodic boundary condition is used to formulate the optic flow estimation problem over the contour.

The stochastic formulation which was adopted for the case of a bounded domain consists in constructing a 2-D stochastic internal model for the optic flow, instead of starting from an arbitrary criterion to be minimized. The 2-D smoother was shown to satisfy a system of elliptic PDEs, which was then solved by using an iterative relaxation method. Experimental results are very encouraging and indicate that this iterative relaxation technique requires only a very short computation time to generate a good estimate of the optic flow inside the domain.

The advantage of the stochastic approach that we have developed here to estimate the optic flow on a contour or inside a domain is that, since it is model-based, it is very flexible. Additional a priori information about the optic flow behavior, or about the object motion can be incorporated easily in the 1-D and 2-D stochastic models that we have constructed. Then, the same procedure can be used to derive the equations satisfied by the estimator. This will be the object of further research.

In addition, since the stochastic estimation approach that we have developed here can be viewed as a model-based regularization technique, it could be applied to a number of problems in computer vision to which regularization methods have been applied in the past, such as robot navigation [20], shape from shading [21], [22], stereo vision [23], etc... The advantage of this approach, as was mentioned above, is the extreme flexibility which exists in generating multidimensional stochastic models.

## REFERENCES

- [1] Poggio, T. and Torre, V., "Ill-Posed Problems and Regularization Analysis in Early Vision", A.I. Lab. Memo No. 773, April 1984, M.I.T., Cambridge MA.
- [2] Horn, B. K. P. and Schunck, B. G., "Determining Optical Flow", *Artificial Intelligence* 17, 1981, pp. 185-203.
- [3] Cornelius, N. and Kanade, T., "Adapting Optical-Flow to Measure Object Motion in Reflectance and X-Ray Image Sequences", *ACM SIGGRAPH/SIGART Interdisciplinary Workshop on Motion: Representation and Perception*, April 1983, Toronto, Canada, pp. 50-58.
- [4] Nagel, H.-H., "Displacement Vectors Derived from Second-Order Intensity Variations in Image Sequences", *Computer Vision, Graphics, and Image Processing* 21, 1983, pp. 85-117.
- [5] Horn, B. K. P., *Robot Vision*, M.I.T. Press, McGraw-Hill, 1986.
- [6] Glazer, F., "Multilevel Relaxation in Low Level Computer Vision", *Multiresolution Image Processing and Analysis*, A. Rosenfeld Ed., Springer-Verlag, 1984, pp. 312-330.
- [7] Terzopoulos, D., "Image Analysis Using Multigrid Relaxation Methods", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-8 No.2, 1986, pp. 129-139.
- [8] Hildreth, E. C., "Computations Underlying the Measurement of Visual Motion", *Artificial Intelligence* 23, 1984, pp. 309-354.
- [9] Bouthémy, P., "Determining Displacements Fields Along Contours From Image Sequences", *Proceedings Vision Interface '86*, Vancouver, pp. 350-355.
- [10] Ljung, L. and Kailath, T., "A Unified Approach to Smoothing Formulas", *Automatica*, Vol. 12 No.2, 1976, pp. 147-157.
- [11] Kailath, T. and Ljung, L., "Two-filter Smoothing Formulas by Diagonalization of the Hamiltonian Equations", *International Journal of Control*, Vol. 36 No.4, 1982, pp. 663-673.

- [12] **Bouthémy, P.**, "A Method of Integrating Motion Information Along Contours Including Segmentation", *Proceedings of the 8th ICPR*, October 1986, Paris.
- [13] **Athans, M. and Falb, P. L.**, *Optimal Control, An Introduction to the Theory and Its Applications* McGraw-Hill, New York, 1966.
- [14] **Bryson, A.E. and Frazier, M.**, "Smoothing for Linear and Nonlinear Dynamic Systems", Aero. Sys. Div., Wright-Patterson Air Force Base, OH, TDR 63-119, 1964, pp. 353-364.
- [15] **Mayne, D. Q.**, "A Solution of the Smoothing Problem for Linear Dynamic Systems", *Automatica* Vol. 4 No.2, 1966, pp.73-92.
- [16] **Fraser, D. C.**, "A New Technique for the Optimal Smoothing of Data", Sc.D. Dissertation, 1967, M.I.T., Cambridge MA.
- [17] **Rauch, H. E., Tung, F. and Striebel, C. T.**, "Maximum Likelihood Estimates of Linear Dynamic Systems", *AIAA Journal* Vol. 3 No.8, 1965, pp. 1445-1450.
- [18] **Adams, M. B., Willsky, A. S. and Levy, B. C.**, "Linear Estimation of Boundary Value Stochastic Processes - Part II: The Role and Construction of Complementary Models", *IEEE Transactions on Automatic Control*, Vol. AC-29 No.9, 1984, pp. 803-811.
- [19] **Kuo, J. C.-C., Levy, B. C., Musicus, B. R.**, "A Local Relaxation Method For Solving Elliptic PDEs on Mesh-Connected Arrays", Report LIDS-P-1508, October 1985, M.I.T., Cambridge MA.
- [20] **Negahdaripour, S. and Horn, B. K. P.**, "Direct Passive Navigation", A.I. Lab. Memo No. 821, February 1985, M.I.T., Cambridge MA.
- [21] **Bruckstein, A. M.**, "On Shape From Shading Methods: Some Theoretical Considerations", Faculty of Electrical Engineering Pub. No. 524, Technion Israel Institute of Technology, Haifa, Israel, June 1985.
- [22] **Ikeuchi, K. and Horn, B. K. P.**, "Numerical Shape From Shading and Occluding Boundaries", *Artificial Intelligence* 7, 1981, pp.141-184.
- [23] **Marr, D. and Poggio, T.**, "A Theory of Human Stereo Vision", *Proceedings of the Royal Society London*, B 204, 1979, pp. 301-328.